

MATLABに関する技術の修得IV

(画像処理と文字認識)

第三技術室システム制御技術班 林 庄司

1. はじめに

MATLABは1980年Moler博士によって開発された。最初はFortran言語を知らない人でも行列計算が対話的にできるようにという目的でMATLAB (Matrix Laboratory) という言語をFortran言語を用いて開発された。1985年にMATLABをC言語化し、機能と品質を大幅にアップして製品化され、工学の分野では圧倒的な顧客を得た。現在第4世代言語とよばれているコンピュータ言語に分類されている。

MATLABの特徴には物理現象および社会現象の記述が簡単にできる、対話的に使え、行列については他に類をみない処理が可能などがある。¹⁾

今回の研修ではMATLABの機能がいくつある中で、MATLAB拡張ソフトである Image_Processing Toolbox⁴⁾ および Neural_Network Toolbox⁵⁾ を用いて、その基本的な使い方の修得を行った。またそれらを用いて、簡単な画像処理と文字認識を行ったので報告する。

2. システム構成とMATLAB言語

2.1 システム構成

MATLABとしては、Windows版MATLAB5.3を用い、パーソナルコンピュータ (V650JD4 飯山電機) にインストールして行った。MATLABソフトとしてインストールされているものは、Table 2.1 に示す通りである。行列計算、簡単な信号処理からSIMULINK, Control_System, Neural_Network Toolboxなど制御問題を解くのに便利なツールやイメージ画像の作成など幅広いツールが備えられている。

またMATLABを使用するための基本的なコマンド、計算などに用いるための演算子、特殊文字については「UNIXのMATLABに関する技術の修得」福井大学技術報告集 Vol3,59-64 (1997) に、関数・

Table 2.1 MATLABソフト一覧

名 称	内 容
MATLAB	数学的計算、工学的計算をするための行列に基づいたシステムで、対話的に物理現象および社会現象を扱う。
SIMULINK	動的な系をシミュレートするための対話型システムで、線形、非線形、連続型、離散型、多変量の系を扱う。
Control_System Toolbox	制御工学ならびにシステム理論によるモデリングの解析、設計を扱う。
Identification Toolbox	システム同定のためのシミュレーションツール。
Signal_Processing Toolbox	1次元、2次元デジタル信号処理のためのもので、デジタル・フィルタの設計と分析、パワー・スペクトル評価を行う。
Image_Processing Toolbox	画像処理をするためのもので、フィルタ処理、画像強調、画像復元、画像解析および統計処理を行うツール。
Neural_Network Toolbox	ニューラルネットワークを用いたシミュレーションツール。

グラフィック等のコマンドは「UNIX版MATLABに関する技術の修得Ⅱ」福井大学技術報告集 Vol4, 45-50 (1998) に、それぞれ記載されているのでここでは省略する。

3. 画像処理とImage Processing

3. 1 画像処理の構成

画像（イメージ）処理は信号処理の中の一分野であり、 今日いろいろな産業・社会で技術革新の原動力となっている。たとえば工学， 医療診断， 天文学， リモートセンシングなどの分野で用いられている。 画像処理はこれからの情報化社会， マルチメディア社会において必須の技術である。

つぎに画像処理構成のためのブロック図は， Fig. 3. 1に示す通りである。 データ取得/保存， 画像情報処理， 濃淡画像処理， 2値画像処理の4つに大別される。

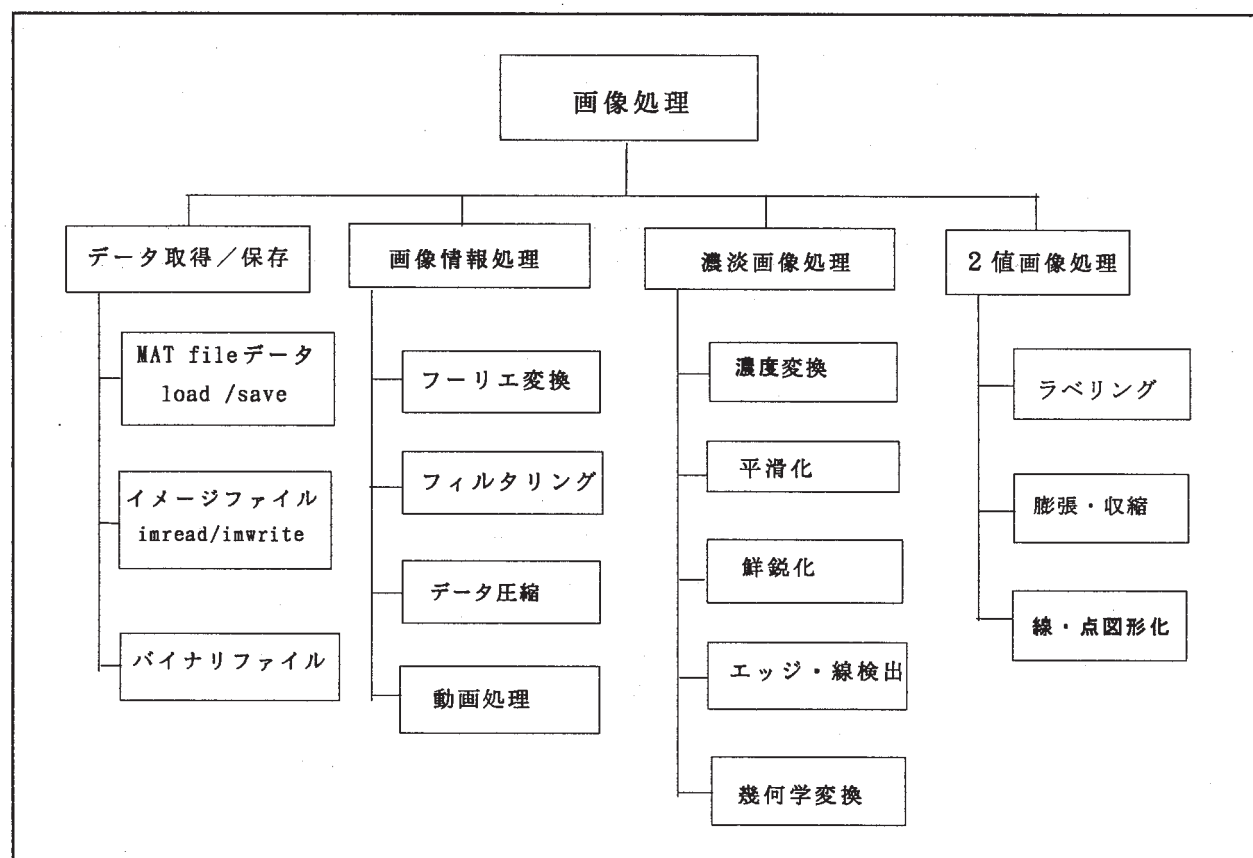


Fig. 3. 1 画像処理のブロック図

これらの詳細については以下に述べる。

データ取得/保存はMatlabで作成されたMATファイルのデータやデジタルカメラ， イメージスキャナで得られた画像データを行列データとして読み書きができ， そのほかCやFortranで得られたバイナリーデータなども読み書きできる。

画像情報処理については， 画像データの空間周波数を解析して処理するものである。これには画像データに離散フーリエ変換を施し， 基本周波数に分解して取り出しを行うことができるフーリエ変換処理がある。このほかフィルタリング（ローパスフィルタ， ハイパスフィルタ）， 画像データを符号化するデータ圧縮， 動画処理がある。

ここでの濃淡画像処理は視覚センサなどを用いて画像認識を行う場合， その前処理としてコントラストの改善， ノイズ除去や図の拡大・縮小を行い， 画像の画質改善を行うものである。この処理方法には濃度変換， 平滑化， 鮮鋭化， エッジ・線検出， 幾何学変換処理がある。

2値画像処理は、文字やCAD図面などの認識を行う場合、0と1（黒と白）の2値画像であると、濃淡画像に比べて情報量が少なく高速処理が可能となる。これにはラベリング、図形を外側に1画素分増減させる膨張・収縮や線・点図形化処理がある。

3. 2 Image_ Processing の構成要素と機能

Image_ Processing Toolboxを用いた画像（イメージ）データ処理は、特殊化された関数をMATLABの計算、グラフィックの環境下で使用できるようにしたものである。そしてそのコマンドおよび関数について、その主なものをTable 3. 1に示す。

Table 3. 1 コマンドおよび関数

	コマンド	説 明
データ入出力 コマンド	load save imread imwrite fread fwrite	MAT Fileの読み込み. MAT Fileの保存 イメージデータの読み込み イメージデータの書き込み CやFortranのデータの読み込み " の書き込み
イメージタイプ 転換関数	ind2gray ind2rgb rgb2gray rgb2hsv edge im2bw	インデックスイメージを強度イメージに変換 " をRGBイメージに変換 RGBイメージを強度イメージに変換 RGBイメージをインデックスイメージに変換 エッジの検出 閾値に基づきイメージをバイナリに変換
幾何学関数	imrotate imcrop imresize imzoom	イメージの回転 イメージの切り取り " の大きさの変更 " の拡大縮小
フィルタ	filter2 fspecial medfilt2 wiener	2次元デジタルフィルタ処理 種々の2次元フィルタの設計 2次元メジアンフィルタリング 2次元適応ノイズ除去フィルタリング
変換関数	fft2 dct2 fft2 dct2 radon	2次元高速フーリエ変換 2次元離散コサイン変換 2次元逆高速フーリエ変換 2次元逆離散コサイン変換 2次元ラドン変換
イメージ グラフィック	imshow colormap colorbar imhist subimage	イメージの描画 カラーマップの設定または復元 カラースケールの設定 イメージヒストグラムの表示 複数のイメージの描画
その他	imnoise subplot bwarea	イメージにノイズを付加 複数のタイル張り表示を制御 バイナリイメージを出力

3.3 画像データ処理

画像データの処理は自動計測、制御工学などの工学の分野では大変重要である。しかしC言語やFortran言語などで画像データを処理をするのは、熟練を要しないと甚だ難しく、またその計算に長時間を要するなどの問題がある。しかしMatlabの拡張ソフトである Image_Processing ツールを用いると、画像処理のための研究、設計、エンジニアリングに対して数値的な計算に必要なものをMファイルとして作成でき、このファイルの変更も簡単に実行することができる。ここでは画像データの処理について述べる。処理手順は、Matlab program 3.3.1 に示す通りである。まず処理画像としてFig.3-3-1 に示すようにノイズを含んだ画像データを作成する。これは模擬データに胡麻塩ノイズを

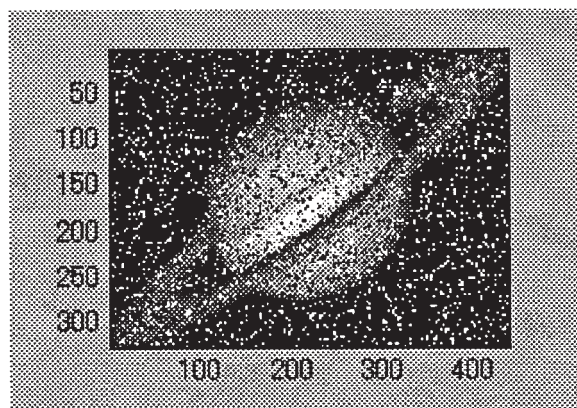


Fig. 3-3-1 ノイズを含んだ画像データ

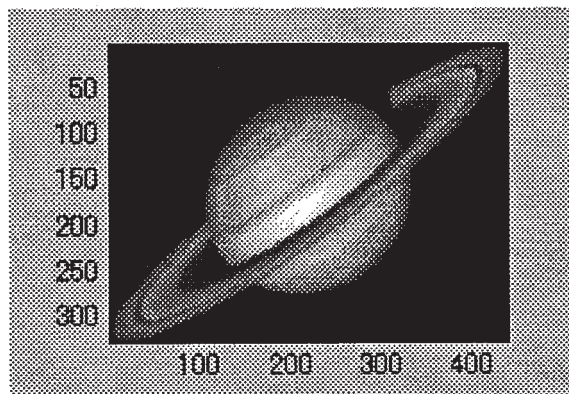


Fig. 3-3-2 フィルタ処理後の画像

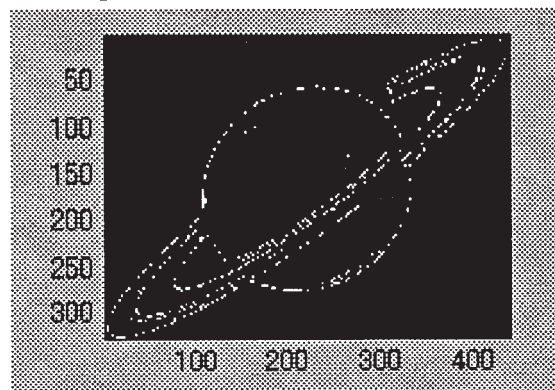


Fig. 3-3-3 エッジ抽出画像

付加して得られた画像データである。このデータをメディアンフィルタを用いて処理を行い画像を復元すると、Fig. 3-3-2のように鮮明な図が得られる。これは5×5近傍の階調値の中央の値を求め、その値を出力画像の各画素値としている。またFig. 3-3-2の画像データを2値化処理することにより、Fig. 3-3-3に示すようにエッジ抽出が行われ、背景画素が黒（0）、図形画素が白（1）の2値化画像が得られる。この図からも明らかに、物体の輪郭を良く抽出していることがわかる。これは文字や輪郭線などの特徴量抽出に良く用いられる手法である。ここでは特に線やエッジの値が大きくなるSobelと呼ばれる差分フィルタを用いている。

Matlab program 3.3.1 (画像処理)

```
%Simulation of Image processing
% Read images and Add noise to an image
I=imread('saturn.tif');
J=imnoise(I,'salt & pepper',0.2);
% Filtering
L=medfilt2(J,[5,5]); % 平滑化 (ノイズ除去)
% Edging
BW=edge(L,'sobel'); % 差分
% Display an image
subplot(3,1,1), subimage(J)
subplot(3,1,2), subimage(L)
subplot(3,1,3), subimage(BW)
```

3.4 文字データの処理と文字認識

ここでは Fig. 3-4-1 に示すイメージ文字データについて2値化処理を行い、ニューラルネットワークにより任意の一文字の認識を行う。認識過程はFig. 3-4-2 のような流れ過程で行い、まず文字

イメージデータを読み込み、雑音ノイズを除去した後、2値化処理を行い任意の一文字を抽出する。そしてFig. 3-4-3に示されるようなニューラルネットワークにより文字認識を行った。ここでのニュー



Fig. 3-4-1 文字イメージデータ

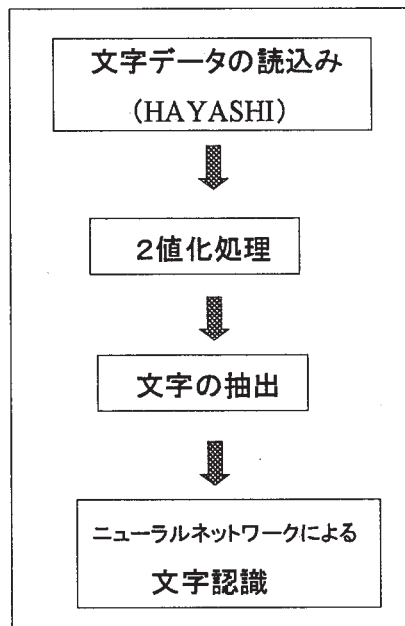


Fig. 3-4-2 文字認識過程

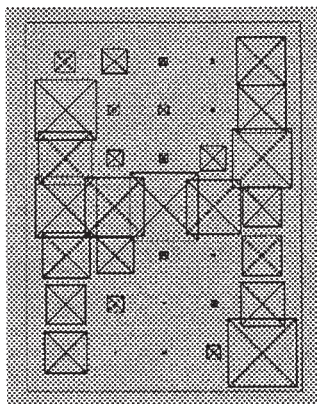


Fig. 3-4-4 ノイズ付加文字

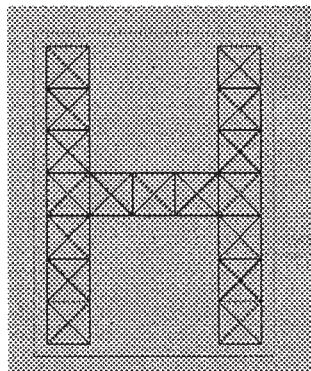


Fig. 3-4-5 認識文字"H"

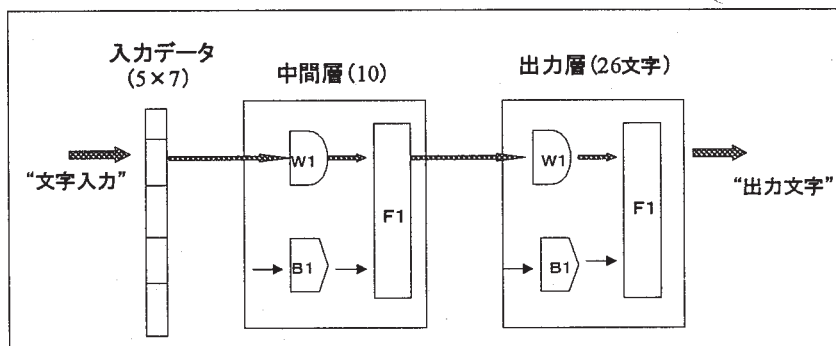


Fig. 3-4-3 ニューラルネットワークによる文字認識

Matlab program 3.4.1 (文字認識)

```

%APPCR1 Character recognition. ( 文字の抽出及び認識 )
echo on
% DEFINING THE MODEL PROBLEM
[alphabet, targets] = prprob;
[R, Q] = size(alphabet);
[S2, Q] = size(targets);

% DEFINING THE NETWORK
S1 = 10;
net = newff(minmax(alphabet), [S1 S2], {'logsig' 'logsig'}, 'traingdx');
net.LW{2,1} = net.LW{2,1}*0.01;
net.b{2} = net.b{2}*0.01;

% TRAINING THE NETWORK WITHOUT NOISE
net.performFcn = 'sse'; % Sum-Squared Error performance function
net.trainParam.goal = 0.1; % Sum-squared error goal.
net.trainParam.show = 20; % Frequency of progress displays (in epochs).
net.trainParam.epochs = 5000; % Maximum number of epochs to train.
net.trainParam.mc = 0.95; % Momentum constant.

% Training begins...please wait...
P = alphabet;
T = targets;
[net, tr] = train(net, P, T);

% TRAINING THE NETWORK
% SET TESTING PARAMETERS
noise_range = 0.3;
max_test = 10;
network1 = [];
network2 = [];
T = targets;

% PERFORM THE TEST
for noiselevel = noise_range
    fprintf('Testing networks with noise level of %.2f.\n', noiselevel);
    errors1 = 0; errors2 = 0;
    for i=1:max_test
        B=imread('HAmoji.bmp'); % 文字の読み込み
        BW=im2bw(B, 0.5); % 文字の二値化
        BW1=imresize(BW, 0.42); % サイズの変更 (拡大, 縮小)
        I=imcrop(BW1, [0 0 5 7]) % 文字の抽出
        b1=I;
        b2=[b1(:,1):b1(:,2):b1(:,3):b1(:,4):b1(:,5):b1(:,6):b1(:,7)]; % データの並替え
        b2=double(b2);
        P = b2 + randn(35, 1)*noiselevel;
        figure(1); plotchar(P);
    % TEST NETWORK 1
    An = sim(net, P);
    AAn = compet(AAn);
    answer = find(AAn == 1);
    figure(2); plotchar(alphabet(:, answer));
    echo off
    end
end
  
```

ラルネットワークは3層の階層型ネットワークで入力層35(5×7), 中間層10, 出力層26のニューロンから構成されている。今回は“H”文字の抽出を行い, その文字にFig. 3-4-4に示されるようなノイズを付加した画像を作成し, ニューラルネットワークにより文字認識を行い, 結果Fig. 3-4-5に示すような文字を得た。図から明らかなように良好に文字認識が行われているのがわかる。ここでのニューラルネットワークの試行回数と学習誤差はFig. 3-4-6に示すようで, 誤差は試行回数200回で0.1であった。

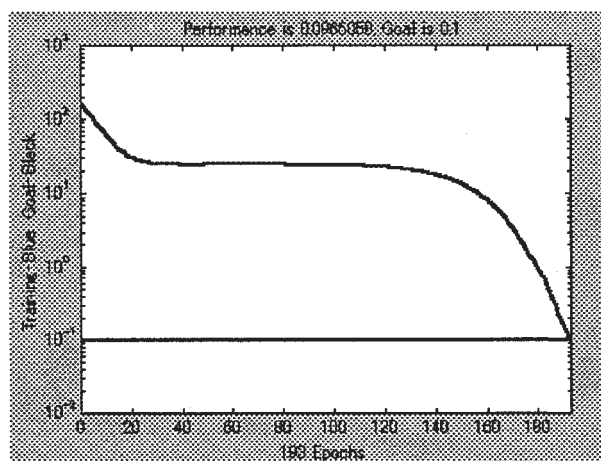


Fig. 3-4-6 学習誤差

4. まとめ

MATLABは物理現象および社会現象のモデル化や記述が簡単にできる利点がある。データ処理や数式処理が対話的に行え, それらの結果をグラフ, 動画として取り出すことができる。制御工学の分野だけではなく, 広く工学の分野でC言語やFortran言語を使用しているユーザに第4世代言語として良く用いられている。

今回の研修ではMATLABの機能が多くある中で, MATLAB拡張ソフトであるToolbox 内にある Image_Processing および Neural_Network のソフトについて, その基本的な使い方の修得を行った。Image_Processing については, デジタルカメラ, イメージスキャナなどから得られた画像データにノイズを付加した模擬画像データを作成し, その画像の復元と特徴量の抽出を行った。そしてこれら一連の処理プログラムの作成を行った。また文字認識については文字データの処理とニューラルネットワークによる文字認識を行い, その処理の有効性を確認できた。よってMATLABのImage_Processing および Neural_Network ソフトの一部分の技術の修得ができた。

5. 今後の課題

MATLAB拡張ソフトでToolbox, SIMULINK内にあり, 今回十分修得できなかったImage_Processing および Neural_Network について, 画像処理, 画像解析などを引き続き行う予定である。またRobust Control, Wavelet, System_Identification解析などのソフトについても理解を深める必要がある。特に非線形問題を解く場合, その系が極めて複雑あるいは数学的に未知な場合についても系の解析が必要とされている。その場合MATLABは解析の有効な手段となりうるし, 大学における研究と学生の技術指導上有用なものである。今後さらに広範囲にMATLAB全般にわたり理解を深め, 一層の技術修得を計る必要がある。

参考文献

- 1) 小国 力 MATLABと利用の実際 1995 サイエンス社
- 2) The MathWorks, Inc. MATLAB User's Guide 1992 サイバネットシステム社
- 3) The MathWorks, Inc. SIMULINK User's Guide 1992 サイバネットシステム社
- 4) Image_Processing Toolbox User's Guide 1997 サイバネットシステム社
- 5) Neural_Network Toolbox User's Guide 1998 サイバネットシステム社